



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/023,379	12/17/2001	Ray M. Jantz	01-777	2447

24319 7590 02/09/2005

LSI LOGIC CORPORATION
1621 BARBER LANE
MS: D-106
MILPITAS, CA 95035

EXAMINER

CLARK, ISAAC R

ART UNIT PAPER NUMBER

2154

DATE MAILED: 02/09/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

10/023,379

Applicant(s)

JANTZ ET AL.

Examiner

Isaac R Clark

Art Unit

2154

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☐ Responsive to communication(s) filed on ____.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-26 is/are pending in the application.
- 4a) Of the above claim(s) ____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) ____ is/are allowed.
- 6) ☒ Claim(s) 1-26 is/are rejected.
- 7) ☐ Claim(s) ____ is/are objected to.
- 8) ☐ Claim(s) ____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 17 December 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. ____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. ____. |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>02/06/2002</u> . | 6) <input type="checkbox"/> Other: ____. |

DETAILED ACTION

1. Claims 1-26 are presented for examination.

Priority

2. The effective filing date for the subject matter in the pending claims in this application is 12/17/2001.

Drawings

3. The Examiner contends that the drawings submitted on 12/17/2001 are acceptable for examination proceedings.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 16 and 23 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

6. Claims 16 and 23 contain the trademark/trade name "Java". Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112, second paragraph. See *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. In the present case, the

Art Unit: 2154

trademark/trade name is used to identify/describe a programming language and a remote method invocation and, accordingly, the identification/description is indefinite.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

8. Claims 1-3, 8-12, and 19-23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dobberpuhl et al. (US Patent 6,754,718, hereinafter Dobberpuhl) in view of Li (US Patent 6,049,829 Li).

9. As per claim 1, Dobberpuhl teaches a computer system comprising a client 130; a host context agent 150; and a storage array, the host context agent providing information to the client to be displayed on the graphical user interface 370 of the client (Fig. 3, Fig. 1; col. 5, lines 18-23).

10. Dobberpuhl does not explicitly teach that the client hosting the graphical user interface is a thin client.

11. Li teaches a graphical user interface on a thin client 100 displaying topological information associated with a storage system (Fig. 2, col. 4, lines 56-59).

12. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li because

Art Unit: 2154

they both deal with managing a storage system and displaying the topology on a client. Furthermore, the teaching of modify the computer system of Dobberpuhl to use a thin client to display the graphical user interface would be a more economical solution because a client having minimal capability is used.

13. As per claim 2, Dobberpuhl teaches the computer system of claim 1, the host context agent (Fig. 1, 150) having a control capability (col. 5, lines 29-33) and comprising a framework for executing code on a corresponding host computer in which the code pushes context information to the storage array from the corresponding host computer (col. 4, lines 60-64) and allows information to be pulled out of the corresponding host computer by the thin client (col. 5, lines 11-22).

14. As per claim 3, Dobberpuhl teaches the computer system of claim 2, wherein the context information includes topology and host type information (col. 3, lines 43-64; col. 5, lines 1-15).

15. As per claim 8, Dobberpuhl teaches the computer system of claim 1, wherein topology acquisition is automated (col. 30-42).

16. As per claim 9, Dobberpuhl teaches the computer system of claim 2, wherein the context information includes host cluster membership (col. 5, lines 4-20).

17. As per claim 10, Dobberpuhl teaches the computer system of claim 2, wherein the control capability includes device registration (col. 5, lines 44-58).

18. As per claim 11, Dobberpuhl teaches the computer system of claim 2, wherein the control capability includes management of services (col. 5, lines 45-

Art Unit: 2154

55: push application periodically updates topology database or allows user to request topology update on demand).

19. As per claim 12, Dobberpuhl teaches the computer system of claim 2, wherein the control capability includes device scanning (col. 4, lines 8-14).

20. As per claim 19, Dobberpuhl teaches a method for host context access in storage array centric storage management interface (Fig. 3), comprising: making a request for host context data on a client (col. 5, lines 1-4); generating and transmitting a provide first host context data command to multiple host computers (Fig. 3, servers 1, 2, 3, and 4); generating and transmitting a provide second host context data command to a storage array (col. 4, lines 1-4).

21. Dobberpuhl fails to explicitly teach that the client is a thin client.

22. Li teaches requesting context information through a thin client 100 (Fig. 2, col. 4, lines 56-59).

23. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li because they both deal with managing a storage system and displaying the topology on a client. Furthermore, the teaching of modify the computer system of Dobberpuhl to use a thin client to request host context data would be provide a cost saving solution because a client having minimal capability is used.

24. As per claim 20, Dobberpuhl teaches the method of claim 19, further comprising generating a first host context data transfer from the host computers to the storage array upon receipt of the first host context data command (col. 4, lines 36-44).

Art Unit: 2154

25. As per claim 21, Dobberpuhl teaches the method of claim 20, further comprising updating the second host context data based on the first host context data (col. 5, lines 10-15).

26. As per claim 22, Dobberpuhl teaches the method of claim 21, further comprising transmitting the second host context data to the thin client (col. 5, lines 20-24).

27. As per claim 23, Dobberpuhl teaches the method of claim 22, further comprising displaying the second host context data (col. 5, lines 20-24).

28. Claims 4-7 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dobberpuhl and Li as applied to claim 1 and 23 above and further in view of further in view of Li et al. (US Published Application, 2002/0143942, hereinafter Li-Alonso).

29. As per claim 4, Dobberpuhl fails to explicitly teach the computer system of claim 1, the host context agent comprising an interface for plugging in host-dependent functions for information gathering and control on a corresponding host computer where a frame work for executing code is running.

30. Li-Alonso teaches a host context agent comprising an interface for interface for plugging in host-dependent functions for information gathering and control on a corresponding host computer where a frame work for executing code is running (See Fig. 1, including wrappers modules for communicating with SAN subsystems and for communicating with clients).

31. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li-Alonso to

Art Unit: 2154

provide a plug-in interface for implementing host dependent functions because they both deal with managing a storage area network via a GUI on a client.

Furthermore, it is well known in the art to partition software into plug-in modules to accommodate different hardware because doing so allows the plug-ins to be developed as needed without affecting the design of the remaining application framework.

32. As per claim 5, Dobberpuhl fails to explicitly teach the computer system of claim 1, the host context agent having plug-in functionality.

33. Li-Alonso teaches a host context agent comprising an interface for interface for plugging in host-dependent functions for information gathering and control on a corresponding host computer where a frame work for executing code is running (See Fig. 1, including wrappers modules for communicating with SAN subsystems and for communicating with clients).

34. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li-Alonso to implement the host context agent using plug-in functionality because they both deal with managing a storage area network via a GUI on a client. Furthermore, it is well known in the art to partition software into plug-in modules to accommodate different hardware because doing so allows the plug-ins to be developed as needed without affecting the design of the remaining application framework.

35. As per claim 6, Dobberpuhl teaches the computer system of claim 1, the host context agent comprising a framework for executing code on a corresponding host computer in which the code pushes context information to the

Art Unit: 2154

storage array from the corresponding host computer (col. 4, lines 60-64) and allows context information to be pulled out of the corresponding host computer by the thin client (col. 5, lines 20-22) wherein the context information includes topology and host type information (col. 3, lines 43-64; col. 5, lines 1-15).

36. Dobberpuhl fails to explicitly teach the host context agent having an interface for plugging in host-dependent functions for information gathering and control on a corresponding host computer where the frame work for executing code is running, and having plug-in functionality.

37. Li-Alonso teaches host context agent having an interface for plugging in host-dependent functions for information gathering and control on a corresponding host computer where the frame work for executing code is running, and having plug-in functionality (See claims 4 and 5 above). The rationale for combining Li-Alonso and Dobberpuhl is as explained for claims 4 and 5 above.

38. As per claim 7, As per claim 7, Dobberpuhl teaches the computer system of claim 6, wherein mapping topology defining ports to hosts are stored in the storage array (col. 3, 60-62: context information includes host ports: col. 5, lines 1-20: database updated to show all current connections and complete topology).

39. Li-Alonso teaches a host architecture supporting modules plugged into the host application (Fig. 3, Paragraph 0025, 0026).

40. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li-Alonso to provide plug-in modules to contain the software necessary to interface with

Art Unit: 2154

varies clients and storage hardware because they both deal with managing a storage area network via a GUI on a client. Furthermore, it is well known in the art to partition software into plug-in modules to accommodate different hardware because doing so allows the plug-ins to be developed as needed without affecting the design of the remaining application framework.

41. As per claim 26, Dobberpuhl fails to explicitly teach the method of claim 23, the method employing Java's Remote Method Invocation as the method for thin-client-to-host communication.

42. Li-Alonso teaches employing remote method invocation for client-to-host communication (Fig. 3, items 322 and 318).

43. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li-Alonso because they both deal with managing a storage area network via a GUI on a client. Furthermore, the teaching of Li-Alonso to use remote method invocation would allow the use of a standard technique for invoking operations between a client and server on different platforms thus facilitating the design of the client application and allowing the host interface to accommodate a variety of client.

44. Claims 13-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dobberpuhl et al. (US Patent 6,754,718, hereinafter Dobberpuhl) in view of Official Notice.

45. As per claim 13, Dobberpuhl teaches a recording medium readable by a computer in which a program is stored (col. 6, lines 56-65), the program for information from an information processing apparatus to an external apparatus

Art Unit: 2154

comprising the steps of: generating and sending a command for a host context information to a host computer having the host context information (col. 4, lines 36-44); and generating and second a command to a storage array for host context information (col. 4, lines 1-4).

46. Dobberpuhl teaches that the information is display information rather than printing information however the use of printers to display a hard copy of information is well known in the art. It would have been obvious at the time the invention was made to modify the program taught by Dobberpuhl to format and send the host context information to a printer to provide a historical record of the storage network topology as an aid to managing the system.

47. As per claim 14, Dobberpuhl teaches the recording medium of claim 13, further comprising receiving the host context information from the storage array (col. 5, lines 18-22).

48. As per claims 15 and 16, Dobberpuhl teaches the recording medium of claim 14, further comprising displaying the host context information on a graphical user interface (col. 5, lines 18-22). Dobberpuhl does not explicitly teach that the computer program is written in the JAVA language, however it does teach that the program can be written in a variety of languages for many architectures and operating systems (col. 7, lines 4-9). It would have been obvious to one of ordinary skill at the time the invention was made to use the JAVA language because the use of JAVA would allow host independent portions of the application to run on a variety of host hardware.

Art Unit: 2154

49. Claims 17, 18, 24, and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dobberpuhl and Official Notice as applied to claim 15 and 23 above and further in view of Li et al. (US Published Application, 2002/0143942, hereinafter Li-Alonso).

50. As per claims 17 and 18 Dobberpuhl fails to explicitly teach recording medium of claim 15, the computer program interfacing with host context agent framework that uses plug ins, and the host context agent framework being a Remote Procedure Call (RPC) server and the plug-ins being RPC procedures

51. Li-Alonso teaches Li-Alonso teaches a host context agent comprising an interface for interface for plugging in host-dependent functions for information gathering and control on a corresponding host computer where a frame work for executing code is running (See Fig. 1, including wrappers modules for communicating with SAN subsystems and for communicating with clients).

52. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li-Alonso to implement the host context agent using plug-in functionality because they both deal with managing a storage area network via a GUI on a client. Furthermore, it is well known in the art to partition software into plug-in modules to accommodate different hardware because doing so allows the plug-ins to be developed as needed without affecting the design of the remaining application framework. Further the use of RPC client-server architectures to implement plug-in functionality between platforms is well known in the art at the time the invention was made. It would have been obvious to implement the host context agent as a

Art Unit: 2154

server and the plug-ins as RPC procedures because using the RPC architecture the program would facilitate the design of the system by partitioning the host specific and peripheral specific functionality in separate entities.

53. As per claims 24 and 25, Dobberpuhl fails to explicitly teach the method of claim 23, the method being on a host context agent framework that uses plug ins, the host context agent framework being a Remote Procedure Call (RPC) server and the plug-ins being RPC procedures.

54. Li-Alonso teaches Li-Alonso teaches a host context agent comprising an interface for interface for plugging in host-dependent functions for information gathering and control on a corresponding host computer where a frame work for executing code is running (See Fig. 1, including wrappers modules for communicating with SAN subsystems and for communicating with clients).

55. It would have been obvious to one of ordinary skill in this art at the time the invention was made to combine the teaching of Dobberpuhl and Li-Alonso to implement the host context agent using plug-in functionality because they both deal with managing a storage area network via a GUI on a client. Furthermore, it is well known in the art to partition software into plug-in modules to accommodate different hardware because doing so allows the plug-ins to be developed as needed without affecting the design of the remaining application framework.

Further the use of RPC client-server architectures to implement plug-in functionality between platforms is well known in the art at the time the invention was made. It would have been obvious to implement the host context agent as a server and the plug-ins as RPC procedures because using the RPC architecture

Art Unit: 2154

the program would facilitate the design of the system by partitioning the host specific and peripheral specific functionality in separate entities.

Conclusion

56. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. The following patents and publications are cited to further show the state of the art with respect to "Method for improved host context access in storage-array-centric storage management interface".

- | | | | |
|-----|--------------|-----------------------|---|
| i. | US 6,769,022 | DeKoning et al. | Managing |
| | | | storage devices and display of topology |
| ii. | US 6,098,128 | Velez-McCaskey et al. | Modular |
| | | | design of storage management system |

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Isaac R Clark whose telephone number is (571)272-3961. The examiner can normally be reached on Monday-Friday 8:00am-4:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John A Follansbee can be reached on (571)272-3964. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Art Unit: 2154

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

IRC

JOHN FOLLANSBEE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100